

**ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД УКООПСПІЛКИ  
«ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ»**

**ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ ТА ІНФОРМАЦІЙНИХ  
ТЕХНОЛОГІЙ  
ФОРМА НАВЧАННЯ ЗАОЧНА  
КАФЕДРА МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ТА СОЦІАЛЬНОЇ  
ІНФОРМАТИКИ**

**Допускається до захисту**

Завідувач кафедри \_\_\_\_\_ О.О. Ємець  
(підпис)

«\_\_\_\_\_» \_\_\_\_\_ 2019 р.

**ПОЯСНЮВАЛЬНА ЗАПИСКА  
ДО ДИПЛОМНОЇ РОБОТИ**

**на тему**

**РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТРЕНАЖЕРУ З ТЕМИ  
«ПРАМІДАЛЬНЕ СОРТУВАННЯ» ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО  
КУРСУ «АЛГОРИТМИ ТА СТРУКТУРИ ДАНИХ»**

**зі спеціальності 122 «Комп'ютерні науки»**

**Виконавець роботи** Самборська Ксенія Юріївна

\_\_\_\_\_ «\_\_\_» \_\_\_\_\_ 2019р.  
(підпис)

**Науковий керівник** к.ф.-м.н., доц. Ємець Олександра Олегівна

\_\_\_\_\_ «\_\_\_» \_\_\_\_\_ 2019р.  
(підпис)

**ПОЛТАВА 2019р.**

## ЗМІСТ

ВСТУП .....	3
1. ПОСТАНОВКА ЗАДАЧІ.....	5
2. ІНФОРМАЦІЙНИЙ ОГЛЯД.....	7
2.1. Огляд робіт з аналогічним завданням.....	7
2.2. Позитивні аспекти оглянутих робіт .....	10
2.3. Вад розробок з оглянутих робіт .....	10
2.4. Необхідність та актуальність теми роботи.....	10
3. ТЕОРЕТИЧНА ЧАСТИНА .....	12
3.1. Огляд матеріалу за темою роботи .....	12
3.2. Алгоритмізація за темою роботи.....	16
3.3. Розробка блок-схеми, яка підлягає програмуванню .....	33
3.4. Обґрунтування вибору програмних засобів для реалізації завдання роботи.....	36
4. ПРАКТИЧНА ЧАСТИНА .....	39
4.1. Опис процесу програмної реалізації .....	39
4.2. Опис програми.....	42
4.3. Необхідна користувачу програми інструкція .....	46
ВИСНОВКИ.....	51
СПИСОК ЛІТЕРАТУРИ.....	53
ДОДАТОК А. КОД ПРОГРАМИ .....	54

## ВСТУП

Для того, щоб студент зміг успішно виконувати свої функції у професійній діяльності, йому необхідно під час навчання набути певних практичних навичок та теоретичних знань. Однією із важливих умов підвищення розвитку теорії і практики є створення системи завдань, їх послідовність та різноманітність.

Метою роботи є розробка програмного забезпечення тренажеру з теми «Пірамідальне сортування» дистанційного начального курсу «Алгоритми та структури даних» та закріплення знань із застосування пірамідального сортування.

Об'єктом розробки є процес дистанційного навчання математичним дисциплінам.

Предметом розробки є програмний продукт, що реалізує тренажер для закріплення знань із застосування пірамідального сортування.

Головне завдання – розробити алгоритм роботи тренажеру з теми «Пірамідальне сортування» і його програмну реалізацію.

Методи розробки – методика застосування пірамідального сортування, мова програмування Java, середовище візуальної розробки програм NetBeans.

Розроблений тренажер надає користувачу можливість перемкнути мову з української на англійську і навпаки.

Тренажер готовий до використання в дистанційному курсі «Алгоритми та структури даних».

Робота складається з чотирьох розділів. У першому розділі розглянуто постановку задачі для реалізації тренажера. У другому розділі описано огляд робіт з аналогічним завданням, позитивні аспекти оглянутих робіт і їх вади розробок, необхідність та актуальність теми роботи. У третьому розділі представлено огляд матеріалу за темою роботи, описано алгоритмізацію, представлено блок-схему алгоритму,

обґрунтування вибору програмних засобів для реалізації завдання роботи. У четвертому розділі – представлено опис процесу програмної реалізації, програми, необхідну користувачу програми інструкцію.

Обсяг пояснювальної записки: 73 стор., в т.ч. основна частина - 47 стор., джерела - 10 назв.

## 1. ПОСТАНОВКА ЗАДАЧІ

Основним завданням роботи є розробка програмного забезпечення тренажеру для закріплення знань із застосування пірамідального сортування.

Можна виділити такі завдання роботи:

- розглянути роботи з аналогічним завданням;
- вказати позитивні аспекти оглянутих робіт;
- вказати вади розробок з оглянутих робіт;
- розробити алгоритм тренажеру з теми «Пірамідальне сортування»;
- обґрунтувати вибір програмних засобів для реалізації програми;
- розробити тренажер;
- описати створену програмну реалізацію;
- описати роботу тренажеру.

При програмуванні тренажеру слід розробити можливість студента звернутися до теоретичного матеріалу з теми, що допоможе йому у проходженні. Теоретичний матеріал має також містити приклади, що були використані при розробці алгоритму.

Якщо відповісти не вірно на будь-яке питання, то повинно виводитися відповідне повідомлення. Воно має містити таку інформацію, щоб користувач зрозумів де було допущено помилку.

При розробці графічного представлення тренажеру слід реалізувати наступні основні панелі:

- головна сторінка тренажеру, міститься інформація про назву тренажера, розробника, керівника;
- умова задачі, розроблені кроки відповідно алгоритму;

- кінцевий результат проходження програми, можливість повернутися на головну сторінку або завершити роботу.

Розроблений тренажер має надавати користувачу можливість перемкнути мову з української на англійську і навпаки, оскільки він може бути використаний при викладанні дисципліни «Алгоритми та структури даних» іноземним студентам.

## 2. ІНФОРМАЦІЙНИЙ ОГЛЯД

### 2.1. Огляд робіт з аналогічним завданням

За посиланням <http://www.cs.usfca.edu/~galles/visualization/HeapSort.html> представлено візуалізацію пірамідального сортування (рис. 2.1) [2].

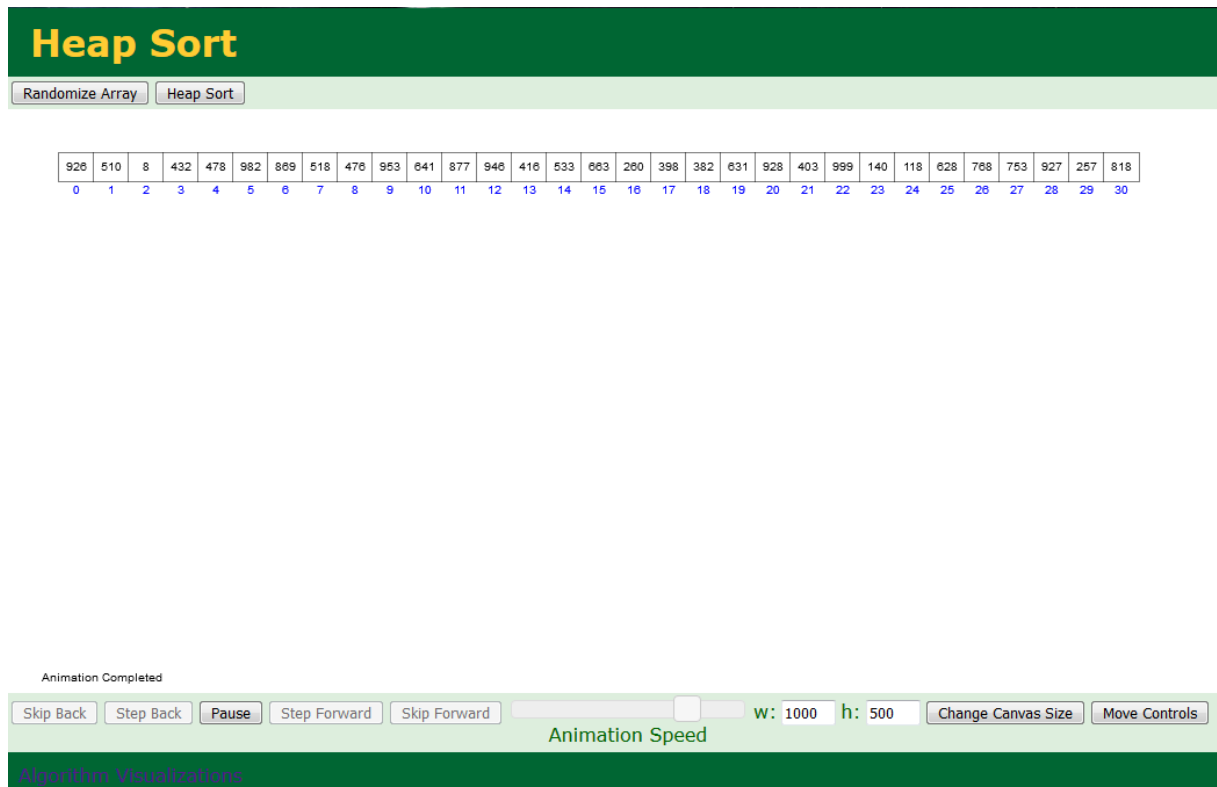


Рисунок 2.2 – Початкова (невідсортована послідовність)

Кнопка "Heap Sort" запускає пірамідальне сортування. "Randomize Array" - розташовує елементи в послідовності у випадковому порядку.

Тут сортування відбувається у порядку зростання, отже у правилі піраміди змінюється знак: елемент  $h_i \geq h_{2i}$  і  $h_i \geq h_{2i+1}$ . У випадку якщо для обох елементів порушується правило піраміди, елемент  $h_i$  міняється місцем з максимальним з елементів  $h_{2i}$  та  $h_{2i+1}$ .

Порівняння елемента  $h_i$  з елементами  $h_{2i}$  та  $h_{2i+1}$ , як видно з анімації, означає порівняння батьківського вузла з його нащадками. Просіювання

елемента, означає його просування по рівнях піраміди - перевірка чи виконується правило піраміди, і якщо не виконується - усунення цього.

Зауважимо, що нумерація в анімації починається з 0, якщо зробити традиційну нумерації, то видно, що дійсно порівнюються елемент  $h_i$  з елементами  $h_{2i}$  та  $h_{2i+1}$  [3].

Розглянемо приклад (рис. 2.2):

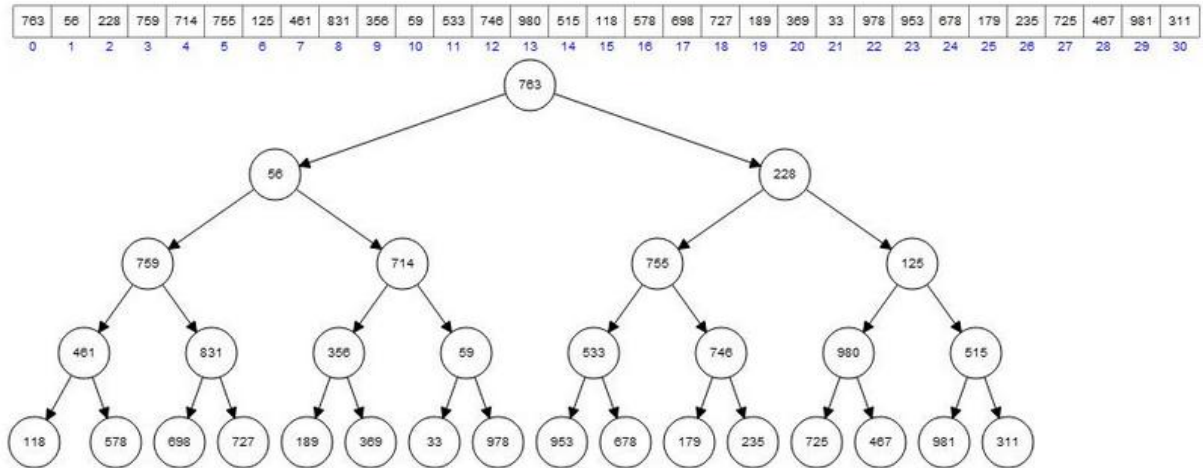


Рисунок 2.2 – Початкова (невідсортована послідовність)

1-ий етап - побудова піраміди, друга половина послідовності, вже є пірамідою, беремо елемент  $h_{14}$  та порівнюємо з  $h_{29}$  та  $h_{30}$  (рис. 2.3).

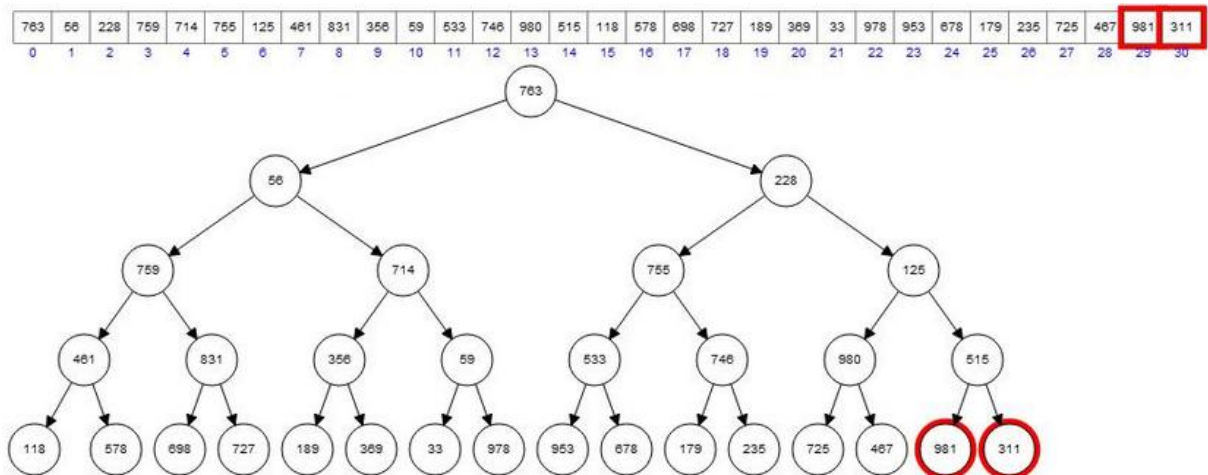


Рисунок 2.3 – Елемент  $h_{14}$  порівнюється з  $h_{29}$  та  $h_{30}$

Для елемента  $h_{14} = 515$  і  $h_{29} = 981$  правило піраміди порушується, отже міняємо їх місцями (рис. 2.4).



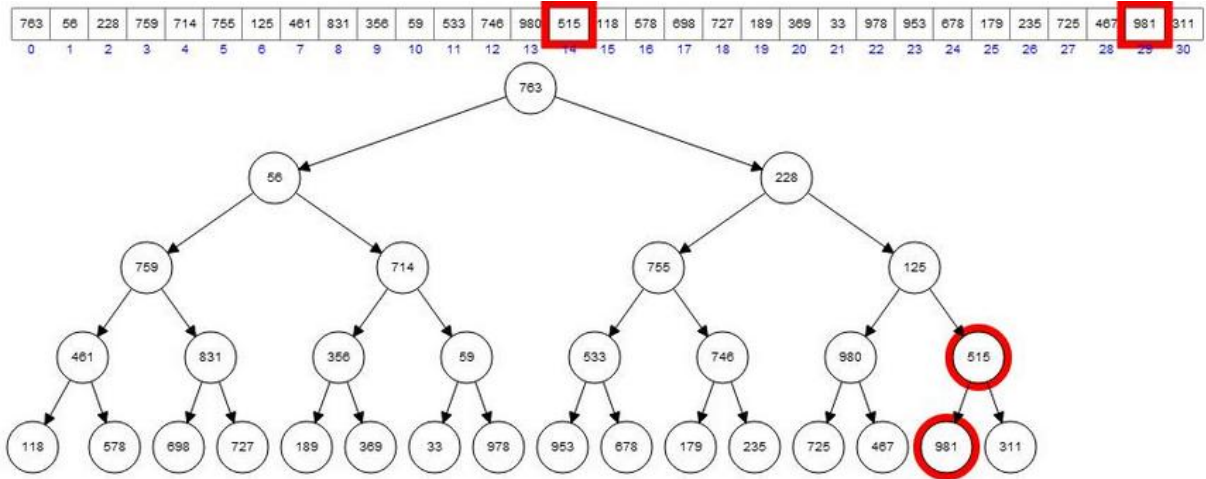


Рисунок 2.4 – Для елемента  $h_{14}$  і  $h_{29}$  правило піраміди порушується

Беремо наступний елемент  $h_{13} = 980$  і порівнюємо з елементами  $h_{27} = 725$  та  $h_{28} = 467$  і т.д. (рис. 2.5).

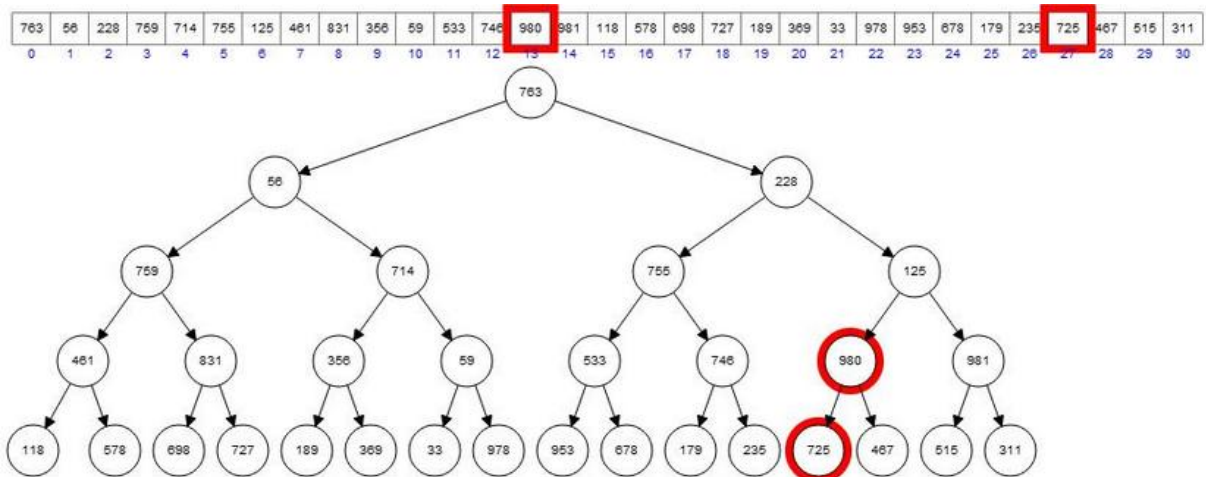


Рисунок 2.5 – Елемент  $h_{13}$  порівнюється з  $h_{27}$  та  $h_{28}$

В кінці отримується відсортована за методом пірамідального сортування послідовність (рис. 2.6).



Рисунок 2.6 – Відсортована послідовність

## **2.2. Позитивні аспекти оглянутих робіт**

Серед позитивних аспектів можна виділити:

- можливість згенерувати послідовність;
- візуальне представлення послідовності у вигляді дерева;
- візуальне представлення кроків побудови піраміди і сортування;
- вибір між автоматичним або поетапним відображенням процесу пірамідального сортування;
- наявність панелі керування процесом.

## **2.3. Води розробок з оглянутих робіт**

Серед вад розробок можна виділити:

- неможливість вказати розмір послідовності;
- відсутня можливість задати власну послідовність;
- доступна лише англійська мова;
- візуалізація пірамідального сортування займає багато часу, якщо більш детально переглядати кожний крок.

## **2.4. Необхідність та актуальність теми роботи**

Сучасні геополітичні зміни та процеси глибинної інформатизації суспільства кардинально змінили умови життя та діяльності людей, їх культуру, світогляд, поведінку тощо. Одним із необхідних компонентів інформатизації освіти є комп'ютер та різноманітні мобільні пристрої [4].

Безперечно, правильне та оптимальне використання інформаційно-комунікаційних технологій дозволяє отримати ряд переваг щодо підвищення ефективності навчально-виховного процесу, а саме:

- забезпечення наочності навчального матеріалу та поєднання різних способів сприйняття навчальної інформації;
- підвищення мотивації за рахунок новизни та різноманітності навчання з використанням комп'ютерної техніки;
- збільшення часу студента для роботи з навчальним матеріалом в індивідуальному режимі;
- наявність об'єктивного контролю та самоконтролю при правильному складанні тестових завдань.

Більшість навчальних та організаційних функцій викладача, які можна перекласти на обчислювальну техніку, реалізуються за допомогою відповідного програмного забезпечення, призначеного для вивчення окремої дисципліни, контролю за діяльністю студентів тощо.

### 3. ТЕОРЕТИЧНА ЧАСТИНА

#### 3.1. Огляд матеріалу за темою роботи

Метод пірамідального сортування поєднує переваги використання деревоподібних структур із сортування на місці, тобто всередині вихідної послідовності. Він належить до групи вдосконалених методів вибору з ефективністю сортування порядку  $n \log_2 n$ .

Пірамідою називається послідовність елементів  $h_1, \dots, h_r$  така, що  $h_i \leq h_{2i}$ ,  $h_i \leq h_{2i+1}$  при  $i = 1, \dots, r/2$ .

Наприклад, послідовність  $h_1=3, h_2=10, h_3=7, h_4=12, h_5=10, h_6=8, h_7=12$  – це піраміда.

Піраміду зручно зображати у вигляді дерева (рис. 3.1).

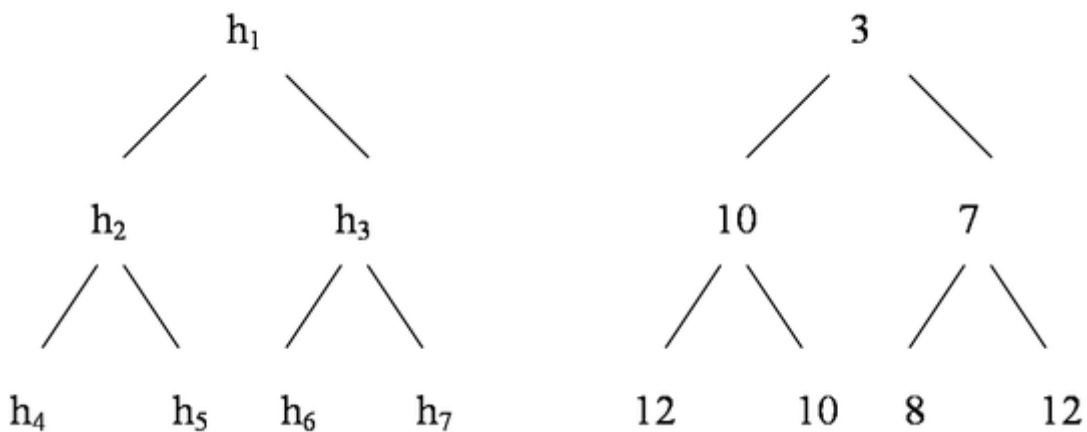


Рисунок 3.1 – Піраміда у вигляді дерева

У вершині піраміди (елемент  $h_1$ ) завжди міститься її мінімальний елемент. Щоб додати елемент до піраміди, не порушуючи її умов, використовують метод, що називається просіванням: елемент  $h_i$ , який додається, послідовно порівнюється з елементами  $h_{2i}$  і  $h_{2i+1}$  й у випадку порушення умови піраміди обмінюється значеннями з меншим із них.

Наприклад, додамо до піраміди (рис. 3.2) елемент  $h_2=9$  (рис. 3.3):

$h_3$	$h_4$	$h_5$	$h_6$	$h_7$	$h_8$
5	2	4	8	7	3

Рисунок 3.2 – Піраміда для прикладу

$h_2$	$h_3$	$h_4$	$h_5$	$h_6$	$h_7$	$h_8$
9	5	2	4	8	7	3




Рисунок 3.3 – Додавання елемента  $h_2$

1.  $h_2 > h_4$  і  $h_2 > h_5$ , з цього випливає, що  $h_2$  обмінюється з  $h_4$  (рис. 3.4);

$h_2$	$h_3$	$h_4$	$h_5$	$h_6$	$h_7$	$h_8$
2	5	9	4	8	7	3




Рисунок 3.4 –  $h_2$  обмінюється з  $h_4$

2. Продовжуємо просівання:  $h_4 > h_8$  (елемент  $h_9$  відсутній, порівнювати ні з чим), з цього випливає, що  $h_4$  обмінюється з  $h_8$ . Отримуємо нову піраміду (рис. 3.5).

$h_2$	$h_3$	$h_4$	$h_5$	$h_6$	$h_7$	$h_8$
2	5	3	4	8	7	9

Рисунок 3.5 – Отримана піраміда

Метод пірамідального сортування включає два основні етапи (сортуємо за спаданням).

1. Побудова піраміди з вихідної послідовності  $a_1, \dots, a_n$ : елементи  $a_{n/2+1}, \dots, a_n$  вже утворюють піраміду; далі кожний з елементів  $a_{n/2}, a_{n/2-1}, \dots, a_1$  просівається на своє місце, послідовно розширюючи піраміду на один елемент вліво. Отримана в результаті цього нова послідовність  $a_1, \dots, a_n$  буде пірамідою.

2. Власне сортування: перший – мінімальний – елемент піраміди міняється з останнім, який просівається на своєму місці в піраміді на одиницю меншого розміру. Знову отримана піраміда розміру  $n-1$  піддається таким самим перетворенням.

Проілюструймо роботу алгоритму на прикладі послідовності 18, 3, 45, -1, 0, 23 ( $n=6$ ).

1. Побудова піраміди.

Елементи  $a_4, a_5, a_6$  вже утворюють піраміду. Розширимо її вліво, просіваючи елемент  $a_3$  (рис. 3.6):

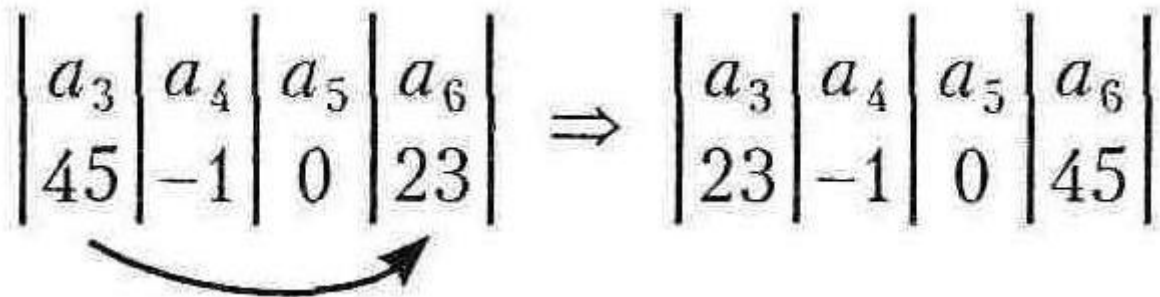


Рисунок 3.6 – Просівання елемента  $a_3$

Аналогічно додаємо елементи  $a_2$  і  $a_1$  (рис. 3.7).

$$\begin{array}{c}
 \left| \begin{array}{c|c|c|c|c|c} a_2 & a_3 & a_4 & a_5 & a_6 & \\ \hline 3 & 23 & -1 & 0 & 45 & \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_2 & a_3 & a_4 & a_5 & a_6 & \\ \hline -1 & 23 & 3 & 0 & 45 & \end{array} \right| \\
 \curvearrowright \\
 \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline 18 & -1 & 23 & 3 & 0 & 45 \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline -1 & 18 & 23 & 3 & 0 & 45 \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline -1 & 0 & 23 & 3 & 18 & 45 \end{array} \right|
 \end{array}$$

Рисунок 3.6 – Додавання елементів  $a_2$  і  $a_1$ 

Отримуємо піраміду. У її вершині – мінімальний елемент усієї послідовності, що дорівнює -1.

## 2. Сортування.

Міняємо місцями перший і шостий елементи, просіваючи  $a_6$  через піраміду  $a_1, \dots, a_5$  (рис. 3.7):

$$\begin{array}{c}
 \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline 45 & 18 & 23 & 3 & 0 & -1 \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline 18 & 45 & 23 & 3 & 0 & -1 \end{array} \right| \\
 \curvearrowright \\
 \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline 23 & 45 & 18 & 3 & 0 & -1 \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline 45 & 23 & 18 & 3 & 0 & -1 \end{array} \right|
 \end{array}$$

Рисунок 3.7 – Просівання  $a_6$  через піраміду  $a_1, \dots, a_5$ 

$a_0$  – другий мінімальний елемент вихідної послідовності. Міняємо  $a_1$  і  $a_5$ , просіваємо через піраміду  $a_1, \dots, a_4$  (рис. 3.8):

$$\left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline 18 & 3 & 23 & 45 & 0 & -1 \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline 3 & 18 & 23 & 45 & 0 & -1 \end{array} \right|$$

Рисунок 3.8 – Зміна  $a_1$  і  $a_5$ , просівання через піраміду  $a_1, \dots, a_4$

Аналогічно продовжуємо процес (рис. 3.9):

$$\begin{array}{c}
 \left| \begin{array}{c|c|c|c|c|c} a_2 & a_3 & a_4 & a_5 & a_6 & \\ \hline 3 & 23 & -1 & 0 & 45 & \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_2 & a_3 & a_4 & a_5 & a_6 & \\ \hline -1 & 23 & 3 & 0 & 45 & \end{array} \right| \\
 \quad \quad \quad \curvearrowright \quad \quad \quad \curvearrowright \\
 \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline 18 & -1 & 23 & 3 & 0 & 45 \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline -1 & 18 & 23 & 3 & 0 & 45 \end{array} \right| \Rightarrow \left| \begin{array}{c|c|c|c|c|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ \hline -1 & 0 & 23 & 3 & 18 & 45 \end{array} \right| \\
 \quad \quad \quad \curvearrowright \quad \quad \quad \curvearrowright
 \end{array}$$

Рисунок 3.9 – Продовження процесу

Останній (точніше, перший) елемент автоматично зайняв своє місце.

Вихідну послідовність відсортовано в порядку спадання (не зростання). Щоб відсортувати її у зворотному порядку (не спадання), слід змінити знак нерівності в умовах піраміди. Тоді у вершині піраміди буде максимальний елемент послідовності [5-7].

### 3.2. Алгоритмізація за темою роботи

На головній сторінці тренажеру міститься інформація про назву тренажера, розробника, керівника. Присутня можливість студента звернутися до теоретичного матеріалу з теми. Надається можливість перемкнути мову з української на англійську і навпаки.

Крок 0. Відсортувати за зростанням вказану послідовність (рис. 3.10). Вибрати дію на поточному етапі:

- Побудувати піраміду.
- Провести сортування.

Якщо вибрано «Побудувати піраміду» – перехід на наступний крок.

Якщо вибрано «Провести сортування» – виводиться повідомлення: «1-ий етап - побудова піраміди!».



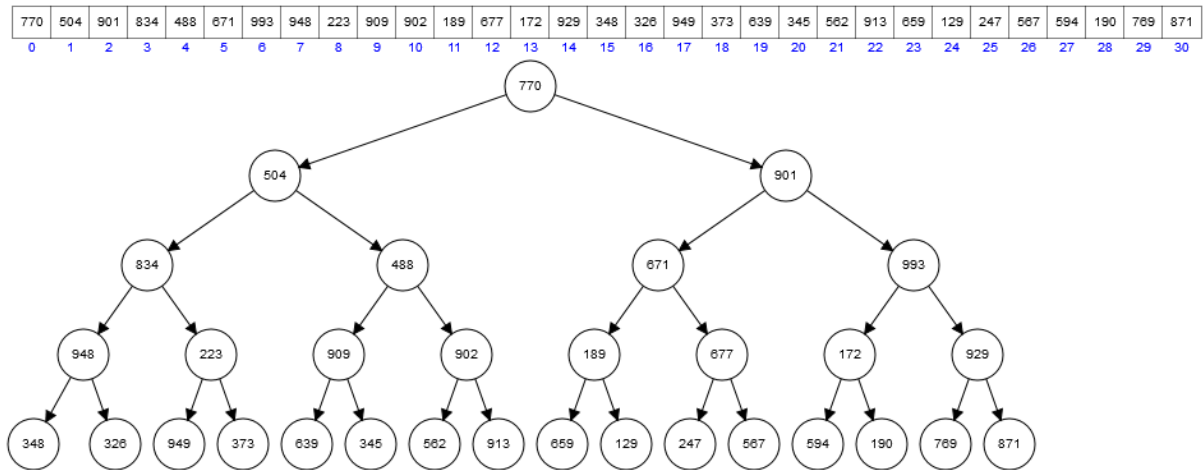


Рисунок 3.10 – Умова задачі

**1-ий етап - побудова піраміди.**

Крок 1. Беремо елемент  $h_{14}$  та порівнюємо з  $h_{29}$  та  $h_{30}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо місцями  $h_{14}$  та  $h_{29}$ .
- Так, порушується. Міняємо місцями  $h_{14}$  та  $h_{30}$ .

Якщо вибрано «Ні, не порушується» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди не порушується!».

Крок 2. Беремо елемент  $h_{13}$  та порівнюємо з  $h_{27}$  та  $h_{28}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо місцями  $h_{13}$  та  $h_{27}$ .
- Так, порушується. Міняємо місцями  $h_{13}$  та  $h_{28}$ .

Якщо вибрано «Так, порушується. Міняємо місцями  $h_{13}$  та  $h_{27}$ » – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо місцями  $h_{13}$  та  $h_{27}$ ».

Крок 3. Беремо елемент  $h_{12}$  та порівнюємо з  $h_{25}$  та  $h_{26}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо місцями  $h_{12}$  та  $h_{25}$ .
- Так, порушується. Міняємо місцями  $h_{12}$  та  $h_{26}$ .

Якщо вибрано «Ні, не порушується» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди не порушується!».

Крок 4. Беремо елемент  $h_{11}$  та порівнюємо з  $h_{23}$  та  $h_{24}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо місцями  $h_{11}$  та  $h_{23}$ .
- Так, порушується. Міняємо місцями  $h_{11}$  та  $h_{24}$ .

Якщо вибрано «Так, порушується. Міняємо місцями  $h_{11}$  та  $h_{23}$ » – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо місцями  $h_{11}$  та  $h_{23}$ ».

Крок 5. Беремо елемент  $h_{10}$  та порівнюємо з  $h_{21}$  та  $h_{22}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо місцями  $h_{10}$  та  $h_{21}$ .
- Так, порушується. Міняємо місцями  $h_{10}$  та  $h_{22}$ .

Якщо вибрано «Так, порушується. Міняємо місцями  $h_{10}$  та  $h_{22}$ » – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо місцями  $h_{10}$  та  $h_{22}$ ».

Крок 6. Беремо елемент  $h_9$  та порівнюємо з  $h_{19}$  та  $h_{20}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо місцями  $h_9$  та  $h_{19}$ .
- Так, порушується. Міняємо місцями  $h_9$  та  $h_{20}$ .

Якщо вибрано «Ні, не порушується» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди не порушується!».

Крок 7. Беремо елемент  $h_8$  та порівнюємо з  $h_{17}$  та  $h_{18}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо місцями  $h_8$  та  $h_{17}$ .
- Так, порушується. Міняємо місцями  $h_8$  та  $h_{18}$ .

Якщо вибрано «Так, порушується. Міняємо місцями  $h_8$  та  $h_{17}$ » – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо місцями  $h_8$  та  $h_{17}$ ».

Крок 8. Беремо елемент  $h_7$  та порівнюємо з  $h_{15}$  та  $h_{16}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо місцями  $h_7$  та  $h_{15}$ .
- Так, порушується. Міняємо місцями  $h_7$  та  $h_{16}$ .

Якщо вибрано «Ні, не порушується» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди не порушується!».

Крок 9. Беремо елемент  $h_6$  та порівнюємо з  $h_{13}$  та  $h_{14}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо  $h_6$  та  $h_{13}$ , просіваємо через піраміду.
- Так, порушується. Міняємо  $h_6$  та  $h_{14}$ , просіваємо через піраміду.

Якщо вибрано «Ні, не порушується» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди не порушується!».

Крок 10. Беремо елемент  $h_5$  та порівнюємо з  $h_{11}$  та  $h_{12}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо  $h_5$  та  $h_{11}$ , просіваємо через піраміду.
- Так, порушується. Міняємо  $h_5$  та  $h_{12}$ , просіваємо через піраміду.

Якщо вибрано «Так, порушується. Міняємо  $h_5$  та  $h_{12}$ , просіваємо через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо  $h_5$  та  $h_{12}$ , просіваємо через піраміду».

Крок 11. Беремо елемент  $h_4$  та порівнюємо з  $h_9$  та  $h_{10}$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо  $h_4$  та  $h_9$ , просіваємо через піраміду.
- Так, порушується. Міняємо  $h_4$  та  $h_{10}$ , просіваємо через піраміду.

Якщо вибрано «Так, порушується. Міняємо  $h_4$  та  $h_{10}$ , просіваємо через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо  $h_4$  та  $h_{10}$ , просіваємо через піраміду».

Крок 12. Беремо елемент  $h_3$  та порівнюємо з  $h_7$  та  $h_8$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо  $h_3$  та  $h_7$ , просіваємо через піраміду.
- Так, порушується. Міняємо  $h_3$  та  $h_8$ , просіваємо через піраміду.

Якщо вибрано «Так, порушується. Міняємо  $h_3$  та  $h_8$ , просіваємо через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо  $h_3$  та  $h_8$ , просіваємо через піраміду».

Крок 13. Беремо елемент  $h_2$  та порівнюємо з  $h_5$  та  $h_6$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо  $h_2$  та  $h_5$ , просіваємо через піраміду.
- Так, порушується. Міняємо  $h_2$  та  $h_6$ , просіваємо через піраміду.

Якщо вибрано «Так, порушується. Міняємо  $h_2$  та  $h_6$ , просіваємо через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо  $h_2$  та  $h_6$ , просіваємо через піраміду».

Крок 14. Беремо елемент  $h_1$  та порівнюємо з  $h_3$  та  $h_4$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо  $h_1$  та  $h_3$ , просіваємо через піраміду.
- Так, порушується. Міняємо  $h_1$  та  $h_4$ , просіваємо через піраміду.

Якщо вибрано «Так, порушується. Міняємо  $h_1$  та  $h_3$ , просіваємо через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо  $h_1$  та  $h_3$ , просіваємо через піраміду».

Крок 15. Беремо елемент  $h_0$  та порівнюємо з  $h_1$  та  $h_2$ . Чи порушується правило піраміди?

- Ні, не порушується.
- Так, порушується. Міняємо  $h_0$  та  $h_1$ , просіваємо через піраміду.
- Так, порушується. Міняємо  $h_0$  та  $h_2$ , просіваємо через піраміду.

Якщо вибрано «Так, порушується. Міняємо  $h_0$  та  $h_2$ , просіваємо через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Правило піраміди порушується! Міняємо  $h_0$  та  $h_2$ , просіваємо через піраміду».

Крок 16. Побудовано піраміду (рис. 3.11). Вибрати дію на поточному етапі:

- Продовжити побудову піраміди.
- Провести сортування.

Якщо вибрано «Провести сортування» – перехід до сортування.

Якщо вибрано «Продовжити побудову піраміди» – виводиться повідомлення: «2-ий етап - сортування!».

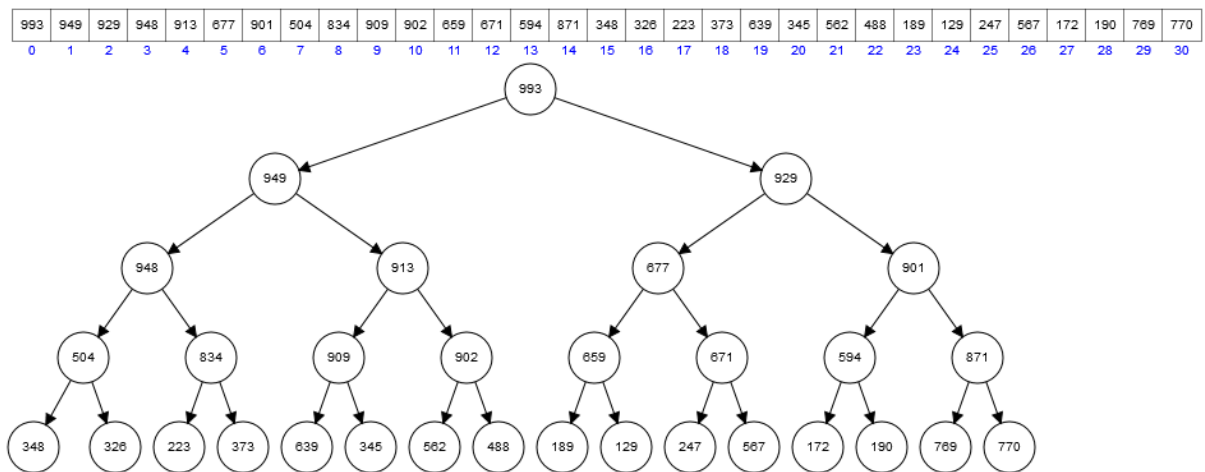


Рисунок 3.11 – Побудована піраміда

## 2-ий етап - сортування.

Крок 1. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований!».

Крок 2. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 3. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований!».

Крок 4. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 5. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований!».

Крок 6. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 7. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.



Якщо вибрано «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований!».

Крок 8. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 9. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований!».

Крок 10. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 11. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований!».

Крок 12. Виберіть дію поточного кроку:

- Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.
- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 13. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.

- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 14. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 15. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 16. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 17. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 18. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 19. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 20. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 21. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 22. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 23. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 24. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 25. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 26. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 27. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 28. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 29. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 30. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 31. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.

- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 32. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 33. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду!».

Крок 34. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».



Крок 35. Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований. Виберіть дію поточного кроку:

- Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.
- Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Якщо вибрано «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду» – перехід на наступний крок.

В іншому разі – виводиться повідомлення: «Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду!».

Крок 36. Послідовність відсортовано за зростанням (рис. 3.12). Надається можливість повернутися на головну сторінку або завершити роботу.

Якщо вибрано «Повернутися на головну сторінку» – перехід на головну сторінку.

129	172	189	190	223	247	326	345	348	373	488	504	562	567	594	639	659	671	677	769	770	834	871	901	902	909	913	929	948	949	993
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

Рисунок 3.12 – Відсортована за зростанням послідовність

### 3.3. Розробка блок-схеми, яка підлягає програмуванню

На рисунку 3.13 зображено блок-схему алгоритму роботи тренажеру.

На рисунках 3.14-3.15 зображено блок-схему процесу «Побудова піраміди».

На рисунках 3.16-3.17 зображено блок-схему процесу «Сортування».

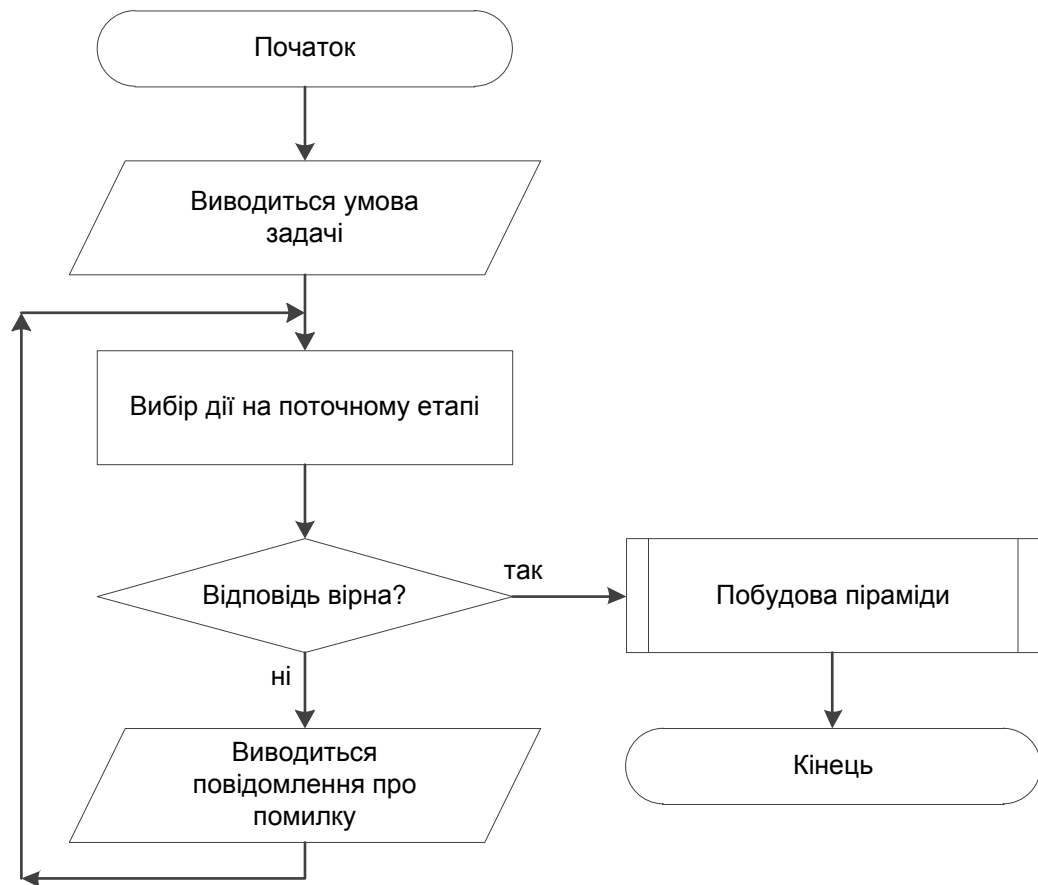


Рисунок 3.13 – Блок-схема алгоритму роботи тренажера

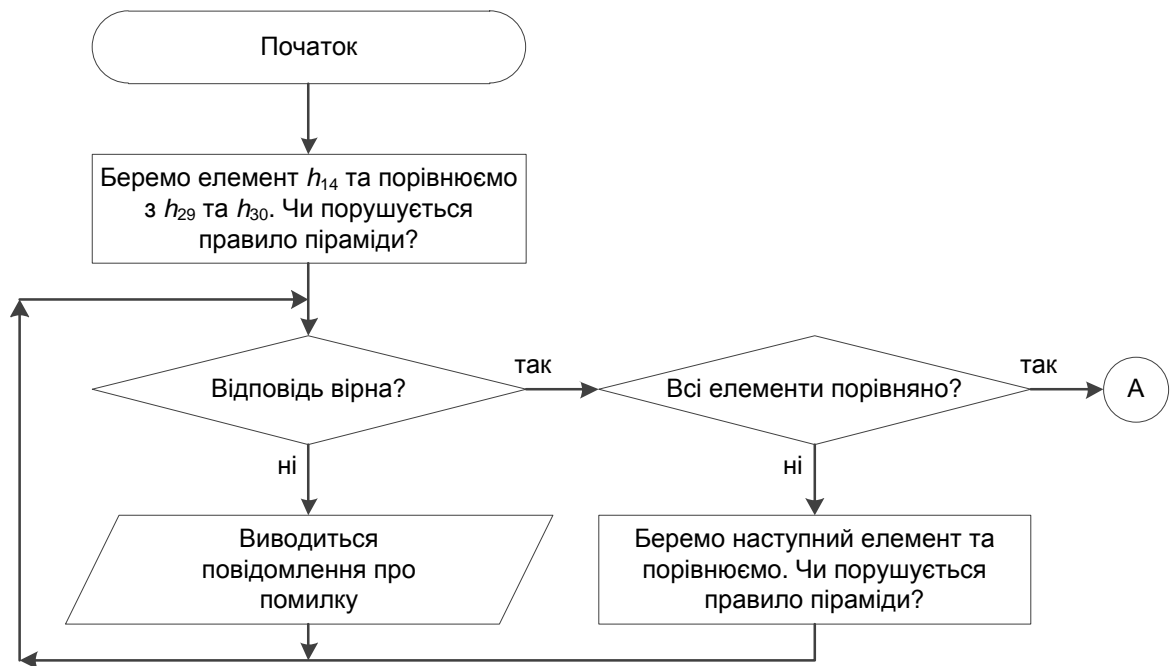


Рисунок 3.14 – Блок-схема процесу «Побудова піраміди»

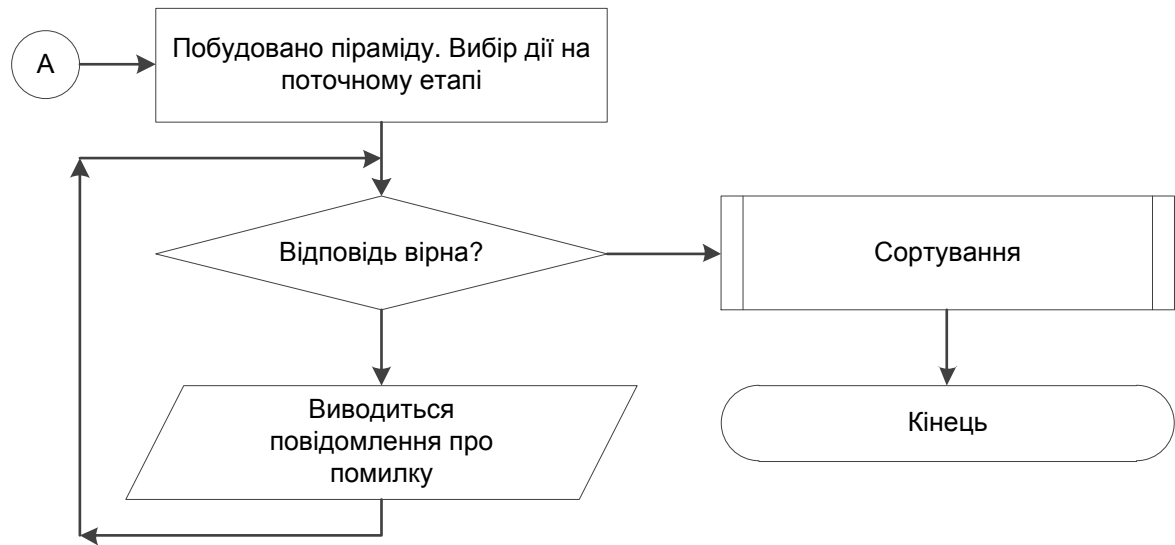


Рисунок 3.15 – Продовження блок-схеми процесу «Побудова піраміди»

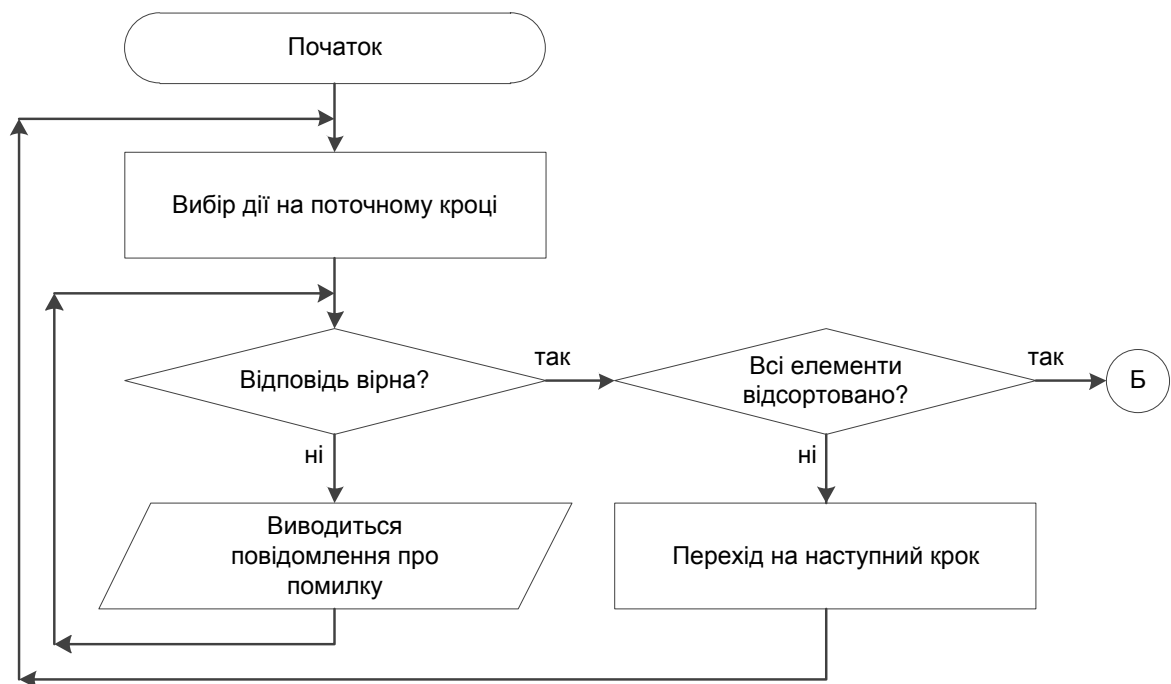


Рисунок 3.16 – Блок-схема процесу «Сортування»

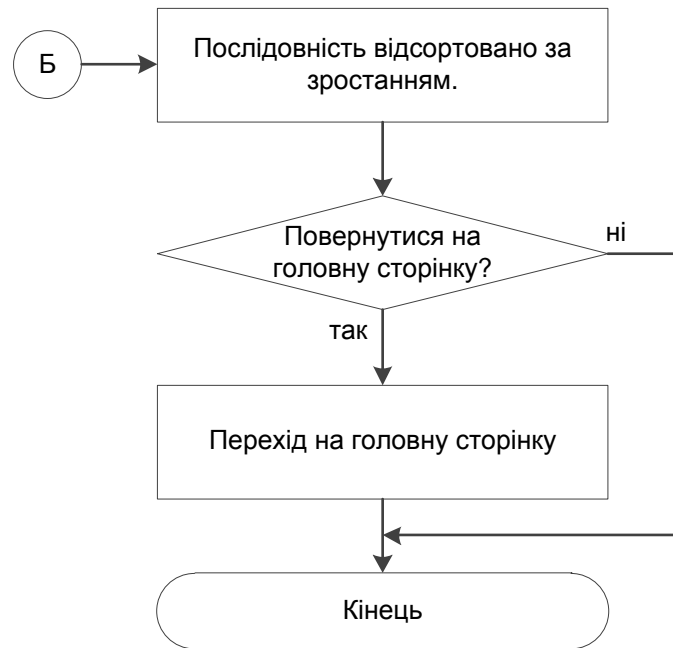


Рисунок 3.17 – Продовження блок-схеми процесу «Сортування»

### 3.4. Обґрунтування вибору програмних засобів для реалізації завдання роботи

Java (вимовляється Джава) — об'єктно-орієнтована мова програмування, випущена 1995 року компанією «Sun Microsystems» як основний компонент платформи Java. З 2009 року мовою займається компанія «Oracle», яка того року придбала «Sun Microsystems». В офіційній реалізації Java-програми компілюються у байт-код, який при виконанні інтерпретується віртуальною машиною для конкретної платформи.

«Oracle» надає компілятор Java та віртуальну машину Java, які задовольняють специфікації Java Community Process, під ліцензією GNU General Public License.

Мова значно запозичила синтаксис із C і C++. Зокрема, взято за основу об'єктну модель C++, проте її модифіковано. Усунуто можливість появи деяких конфліктних ситуацій, що могли виникнути через помилки

програміста та полегшено сам процес розробки об'єктно-орієнтованих програм. Ряд дій, які в C/C++ повинні здійснювати програмісти, доручено віртуальній машині. Передусім Java розроблялась як платформно-незалежна мова, тому вона має менше низькорівневих можливостей для роботи з апаратним забезпеченням, що в порівнянні, наприклад, з C++ зменшує швидкість роботи програм. За необхідності таких дій Java дозволяє викликати підпрограми, написані іншими мовами програмування.

Java вплинула на розвиток J++, що розроблялась компанією «Microsoft». Роботу над J++ було зупинено через судовий позов «Sun Microsystems», оскільки ця мова програмування була модифікацією Java. Пізніше в новій платформі «Microsoft» .NET випустили J#, щоб полегшити міграцію програмістів J++ або Java на нову платформу. З часом нова мова програмування C# стала основною мовою платформи, перейнявши багато чого з Java. J# востаннє включався в версію Microsoft Visual Studio 2005. Мова сценаріїв JavaScript має схожу із Java назву і синтаксис, але не пов'язана із Java [8].

У створенні мови програмування Java було п'ять початкових цілей:

1. Синтаксис мови повинен бути «простим, об'єктно-орієнтовним та звичним».
2. Реалізація має бути «безвідмовною та безпечною».
3. Повинна зберегтися «незалежність від архітектури та переносність».
4. Висока продуктивність виконання
5. Мова має бути «інтерпретованою, багатонитевою, із динамічним зв'язуванням модулів».

NetBeans IDE — вільне інтегроване середовище розробки (IDE) для мов програмування Java, JavaFX, C/C++, PHP, JavaScript, HTML5, Python, Groovy. Середовище може бути встановлене і для підтримки окремих мов, і у повній конфігурації. Середовище розробки NetBeans за замовчуванням підтримує розробку для платформ J2SE і J2EE.

Поширюється у сирцевих текстах під ліцензією Apache License. Проект NetBeans IDE підтримувався і спонсорувався фірмою Sun Microsystems і після придбання Sun — Oracle. У жовтні 2016 року Oracle передав NetBeans у власність Apache Software Foundation, яка займається розробкою і підтримкою проекту.

NetBeans IDE доступна для платформ Microsoft Windows, GNU/Linux, FreeBSD, і Solaris (як SPARC, так x86). Для інших платформ доступна можливість зібрати NetBeans самостійно із сирцевих текстів.

За якістю і можливостям останні версії NetBeans IDE змагається з найкращими інтегрованими середовищами розробки для мови Java, підтримуючи рефакторинг, профілювання, виділення синтаксичних конструкцій кольором, автодоповнення мовних конструкцій на льоту, шаблони коду та інше.

NetBeans IDE підтримує плагіни, дозволяючи розробникам розширювати можливості середовища [9].

## 4. ПРАКТИЧНА ЧАСТИНА

### 4.1. Опис процесу програмної реалізації

Спочатку було створено змінні, які будуть використовуватися:

- panel – використовується для відображення необхідної панелі;
- step – вказується поточний крок алгоритму;
- h – вказується номер елемента заданої послідовності.

```
CardLayout panel;
```

```
int step=0;
```

```
int h=15;
```

Для виведення повідомлення у разі невірної відповіді реалізовано функцію `ShowMessage(String message)`, де текст повідомлення вказується параметром `message`.

```
private void ShowMessage(String message) {
    JLabel label = new JLabel("<html>" + message);
    label.setFont(new Font("Times New Roman", Font.PLAIN, 16));
    JOptionPane.showMessageDialog(Panels, label, "Помилка",
    JOptionPane.WARNING_MESSAGE);
}
```

Для кожної кнопки у тренажері розроблено свою подію, яка буде виконуватися при натисненні кнопки. Так кнопки «Укр» і «Eng» перемикають вміст головної панелі та панелі з теоретичним матеріалом на відповідну мову.

```
private void UkrActionPerformed(java.awt.event.ActionEvent evt) {
    Ukr.setText("<html><u>Укр</u>");
    Ukr.setSelected(true);
    Eng.setText("<html>Eng");
}
```

```

Eng.setSelected(false);
jLabel15.setText("Тренажер з теми: Пірамідальне сортування");
jLabel16.setText("Дистанційний навчальний курс: Алгоритми та
структури даних");
jLabel17.setText("Розробила:          Самборська          К.Ю.
Керівник: Ємець Ол-ра.О.");
Theory.setText("Теоретичний матеріал");
Start.setText("Розпочати");
BuildChoose.setText("Вибрати");
SortChoose.setText("Вибрати");
Back.setText("Назад");
jLabel13.setIcon(new
ImageIcon(getClass().getResource("/images/material.png")));
}
private void EngActionPerformed(java.awt.event.ActionEvent evt) {
    Ukr.setText("<html>Укр");
    Ukr.setSelected(false);
    Eng.setText("<html><u>Eng</u>");
    Eng.setSelected(true);
    jLabel15.setText("Simulator on the theme: Pyramid sorting");
    jLabel16.setText("Distance Learning Course: Algorithms and Data
Structures");
    jLabel17.setText("Developed by: Samborska K.Yu.          Leader:
Yemets Ol-ra.O.");
    Theory.setText("Theoretical material");
    Start.setText("Get started");
    BuildChoose.setText("Select");
    SortChoose.setText("Select");
    Back.setText("Back");

```



```

        jLabel13.setIcon(new
ImageIcon(getClass().getResource("/images/material_eng.png")));
    }

```

Кнопка «Теоретичний матеріал» відображає панель з теоретичним матеріалом.

```

private void TheoryActionPerformed(java.awt.event.ActionEvent evt) {
    panel = (CardLayout) Panels.getLayout();
    panel.show(Panels, "material");
}

```

Кнопка «Назад» здійснює перехід до головної панелі тренажеру.

```

private void BackActionPerformed(java.awt.event.ActionEvent evt) {
    panel = (CardLayout) Panels.getLayout();
    panel.show(Panels, "mainpage");
}

```

При натисненні кнопки «Розпочати» виводиться панель з кроками алгоритму. Також в залежності від вибраної мови вибирається текст деяких написів та варіантів відповіді (див. Додаток А).

```

private void StartActionPerformed(java.awt.event.ActionEvent evt) {
    panel = (CardLayout) Panels.getLayout();
    panel.show(Panels, "build");
}

```

...

Кнопка «Вибрати» перевіряє вибрану відповідь. Якщо вірно – відображається наступний крок алгоритму, якщо ні – виводиться повідомлення за допомогою функції `ShowMessage(String message)`.

Для панелі з процесом побудови піраміди (див. Додаток А):

```
private void BuildChooseActionPerformed(java.awt.event.ActionEvent
evt) {
    boolean check=false;
    switch(step) {
        case 0:
    ...
```

Для панелі з процесом сортування (див. Додаток А):

```
private void SortChooseActionPerformed(java.awt.event.ActionEvent
evt) {
    boolean check=false;
    switch(step) {
        case 1:
    ...
```

## 4.2. Опис програми

На головній панелі тренажеру знаходиться (рис. 4.1):

- тема;
- назва дистанційного курсу;
- прізвище, ініціали розробника;
- прізвище, ініціали наукового керівника;
- перехід до теоретичного матеріалу;
- перехід до умови задачі й подальшого розв'язання;
- вибір мови.



Рисунок 4.1 – Головна панель тренажеру

Панель з теоретичним матеріалом крім інформації містить кнопку «Назад» (рис. 4.2).

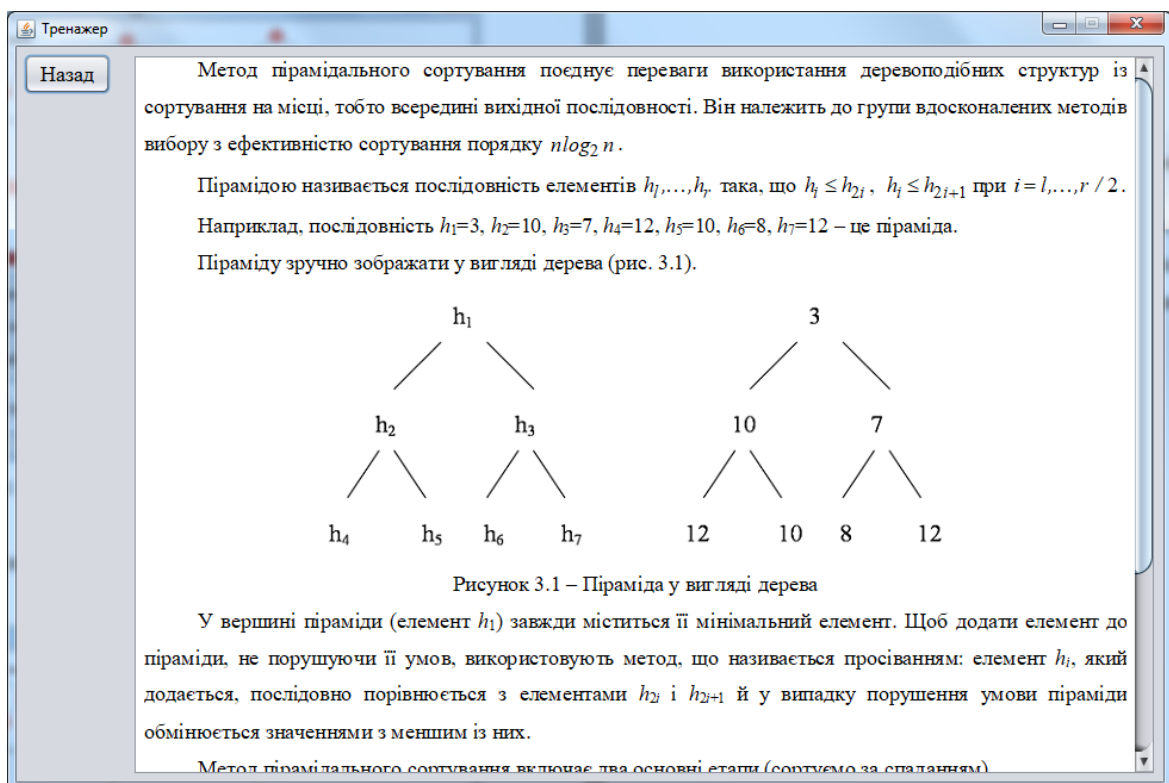


Рисунок 4.2 – Панель з теоретичним матеріалом

Спочатку виводиться панель з початковим кроком алгоритму (рис. 4.3)

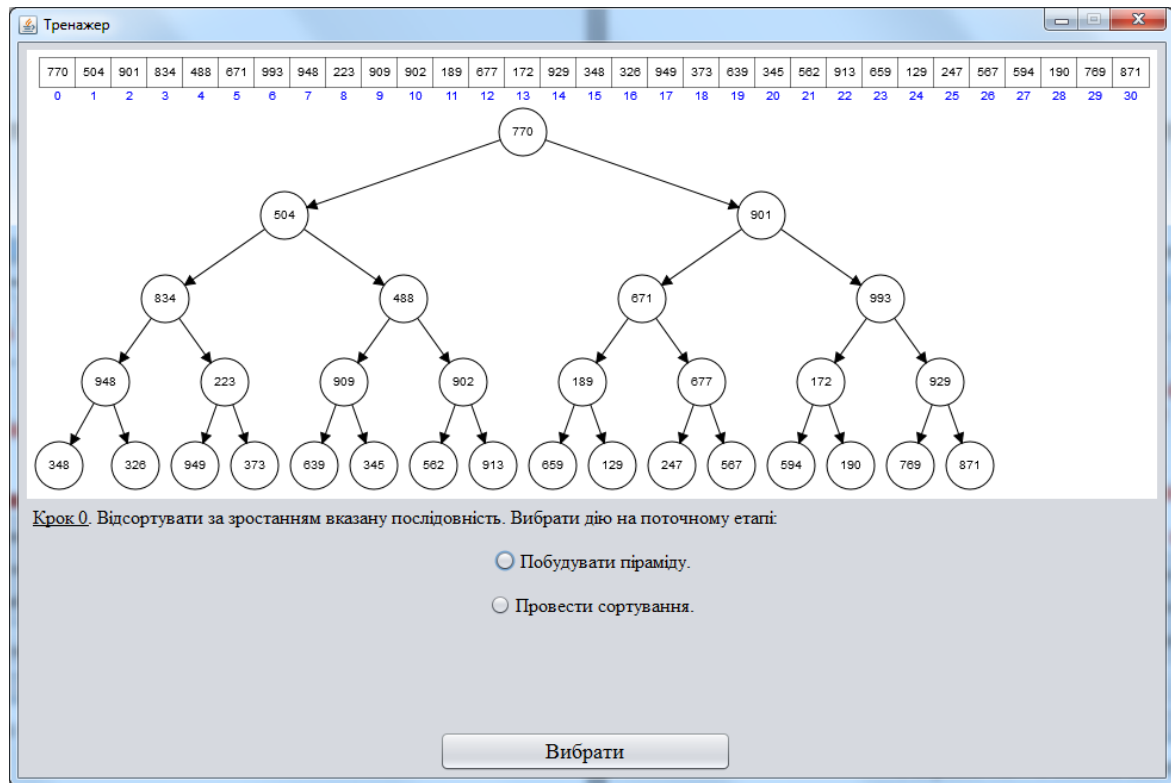


Рисунок 4.3 – Початковий крок алгоритму

Якщо відповідь не вибрано або вибрано не вірно, то виводиться відповідне повідомлення (рис.4.4).

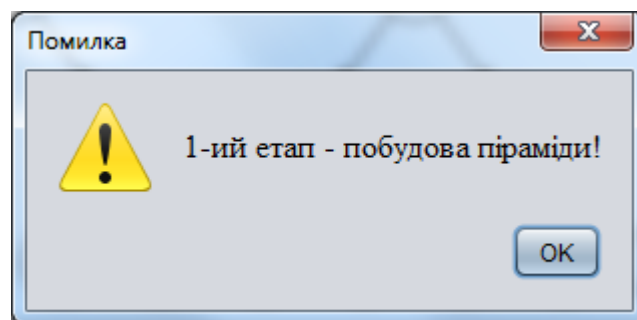


Рисунок 4.4 – Повідомлення про помилку на початковому кроці алгоритму

Потім виводиться панель з процесом побудови піраміди (рис. 4.5). На останньому кроці відбувається перехід до панелі з процесом сортування (рис. 4.6).

Тренажер

770	504	901	834	488	671	993	948	223	909	902	189	677	172	929	348	326	949	373	639	345	562	913	659	129	247	567	594	190	769	871
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

Крок 1. Беремо елемент  $h_{14}$  та порівнюємо з  $h_{29}$  та  $h_{30}$ . Чи порушується правило піраміди?

☐ Ні, не порушується.  
☐ Так, порушується. Міняємо місцями  $h_{14}$  та  $h_{29}$ .  
☐ Так, порушується. Міняємо місцями  $h_{14}$  та  $h_{30}$ .

Вибрати

Рисунок 4.5 – Перший крок алгоритму побудови піраміди

Тренажер

993	949	929	948	913	677	901	504	834	909	902	659	671	594	871	348	326	223	373	639	345	562	488	189	129	247	567	172	190	769	770
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

Крок 1. Виберіть дію поточного кроку:

☐ Міняємо місцями перший і останній елементи. Останній елемент помічаємо як відсортований.  
☐ Міняємо місцями  $h_0$  і  $h_1$ , просіваючи  $h_1$  через піраміду.  
☐ Міняємо місцями  $h_0$  і  $h_2$ , просіваючи  $h_2$  через піраміду.

Вибрати

Рисунок 4.6 – Перший крок алгоритму сортування

Після сортування надається можливість перейти на головну панель або завершити роботу (рис. 4.7).

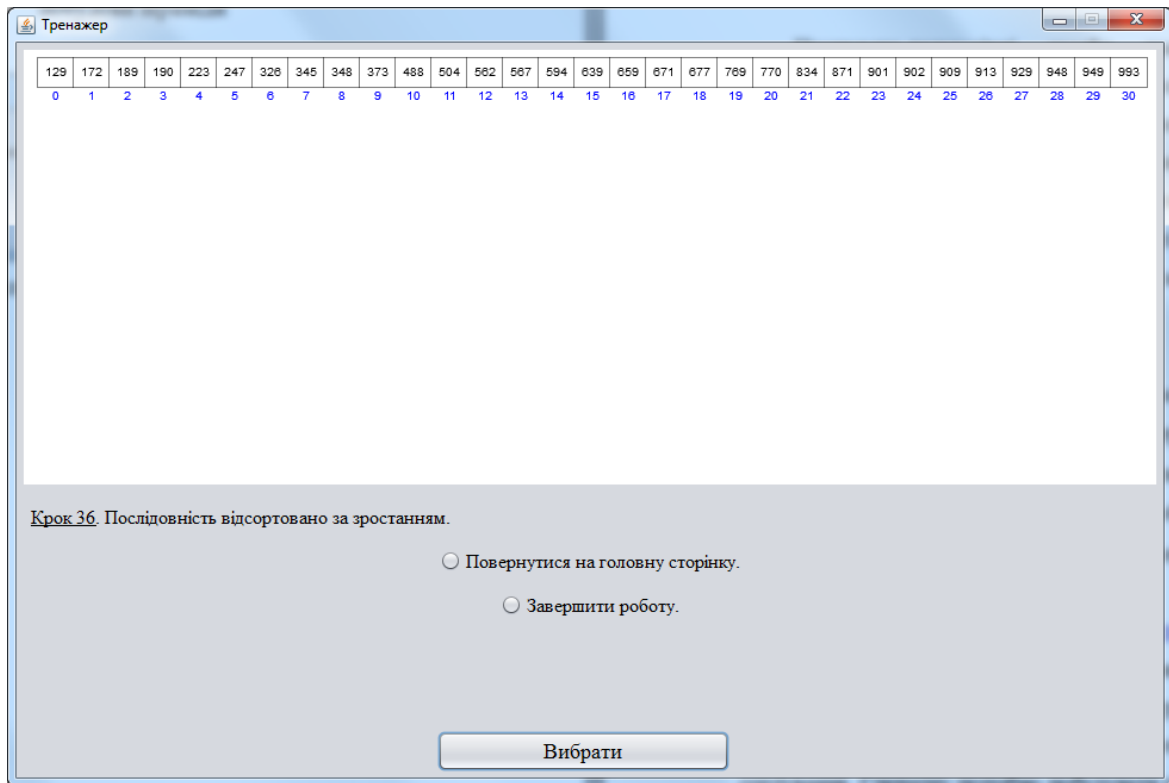


Рисунок 4.7 – Запропоновану послідовність відсортовано

### 4.3. Необхідна користувачу програми інструкція

Щоб ознайомитися з теоретичним матеріалом необхідно натиснути кнопку «Теоретичний матеріал» (рис. 4.2). Для повернення до головної панелі слід натиснути «Назад».

Для зміни мови потрібно вибрати необхідну «Укр» або «Eng». При цьому змінюється вміст головної панелі (рис. 4.8) і теоретичного матеріалу (рис. 4.9).

По замовчування вибрано українську мову.

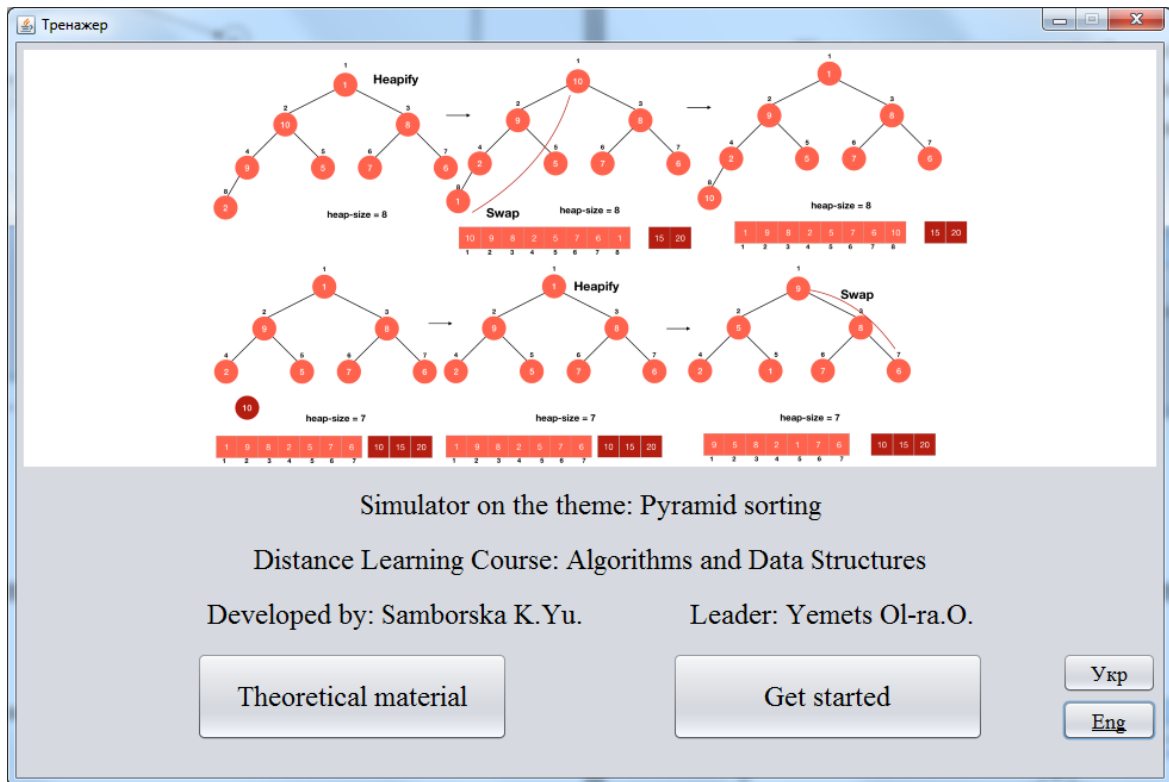


Рисунок 4.8 – Головна панель тренажеру англійською мовою

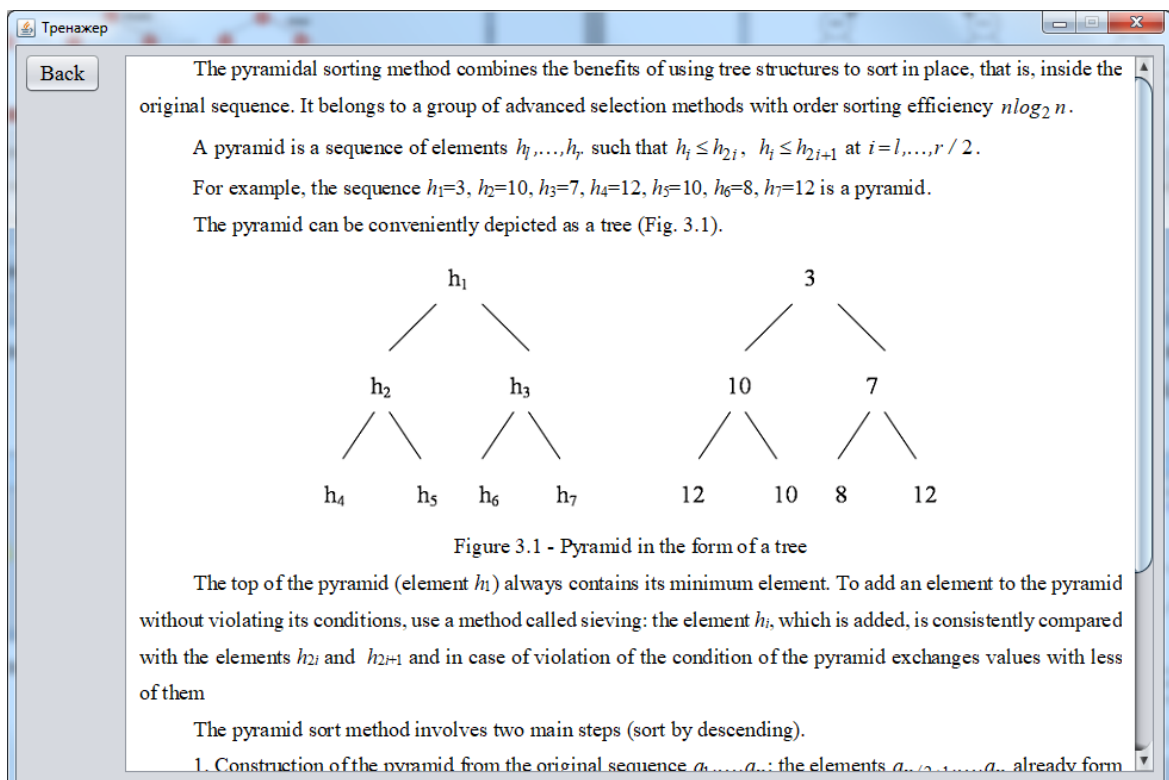


Рисунок 4.9 – Панель з теоретичним матеріалом англійською мовою

Щоб почати виконання тренажеру слід натиснути кнопку «Розпочати». Виведеться панель з початковим кроком алгоритму, потрібно вибрати дію на поточному етапі (рис. 4.3).

Після цього відображається перший крок алгоритму побудови піраміди (рис. 4.5). Якщо відповідь невірна, то виведеться відповідне повідомлення (рис. 4.10). Якщо вірна – перехід до наступного кроку.

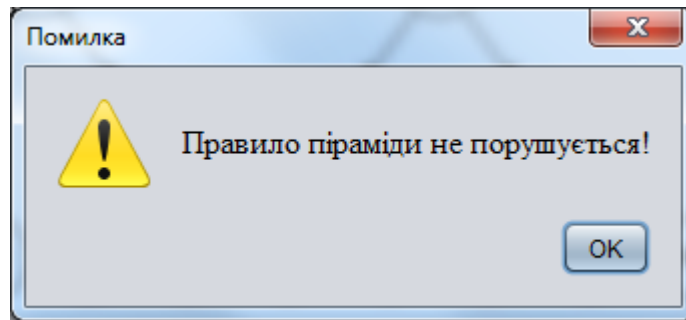


Рисунок 4.10 – Повідомлення про помилку на першому кроці алгоритму побудови піраміди

Після побудови піраміди пропонується вибрати дію на поточному етапі (рис. 4.11).

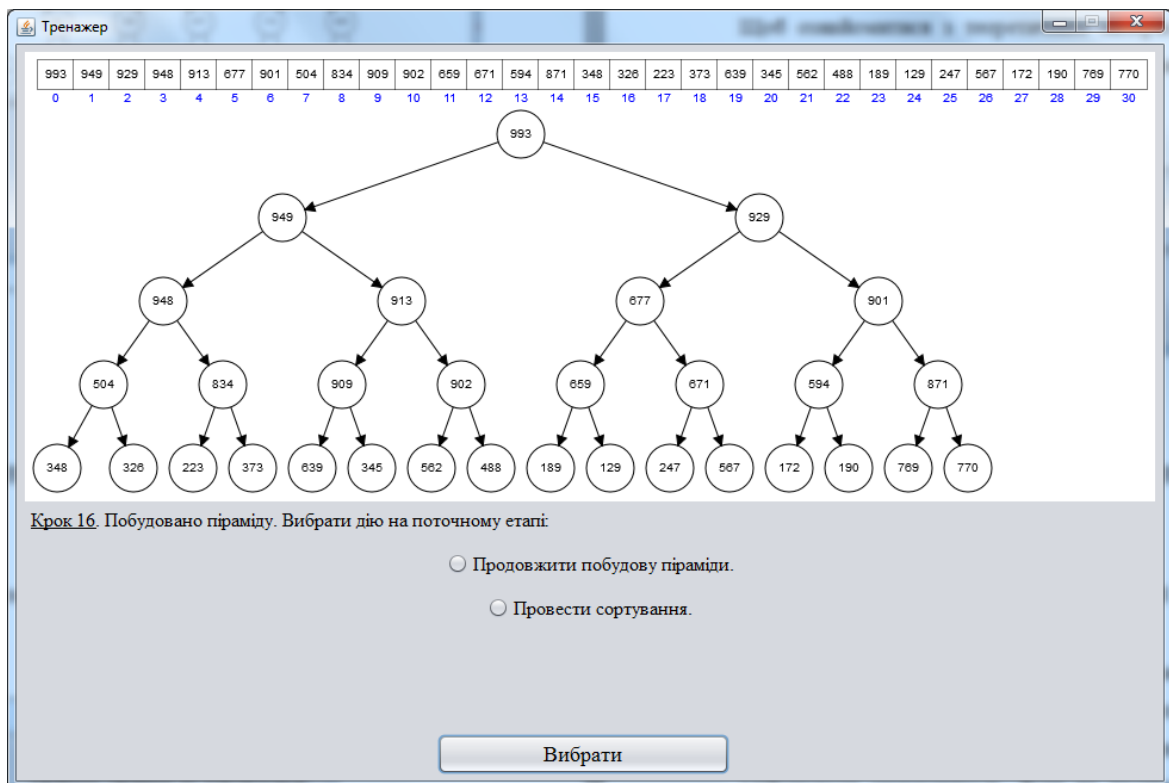


Рисунок 4.11 – Вибір дії після побудови піраміди



Наступний етап – сортування. Відображається панель з першим кроком (рис. 4.6). Після дванадцятого кроку автоматично міняються місцями перший і останній елементи, останній помічається як відсортований. Потрібно вибрати дію із двох варіантів, що залишилися (рис. 4.12).

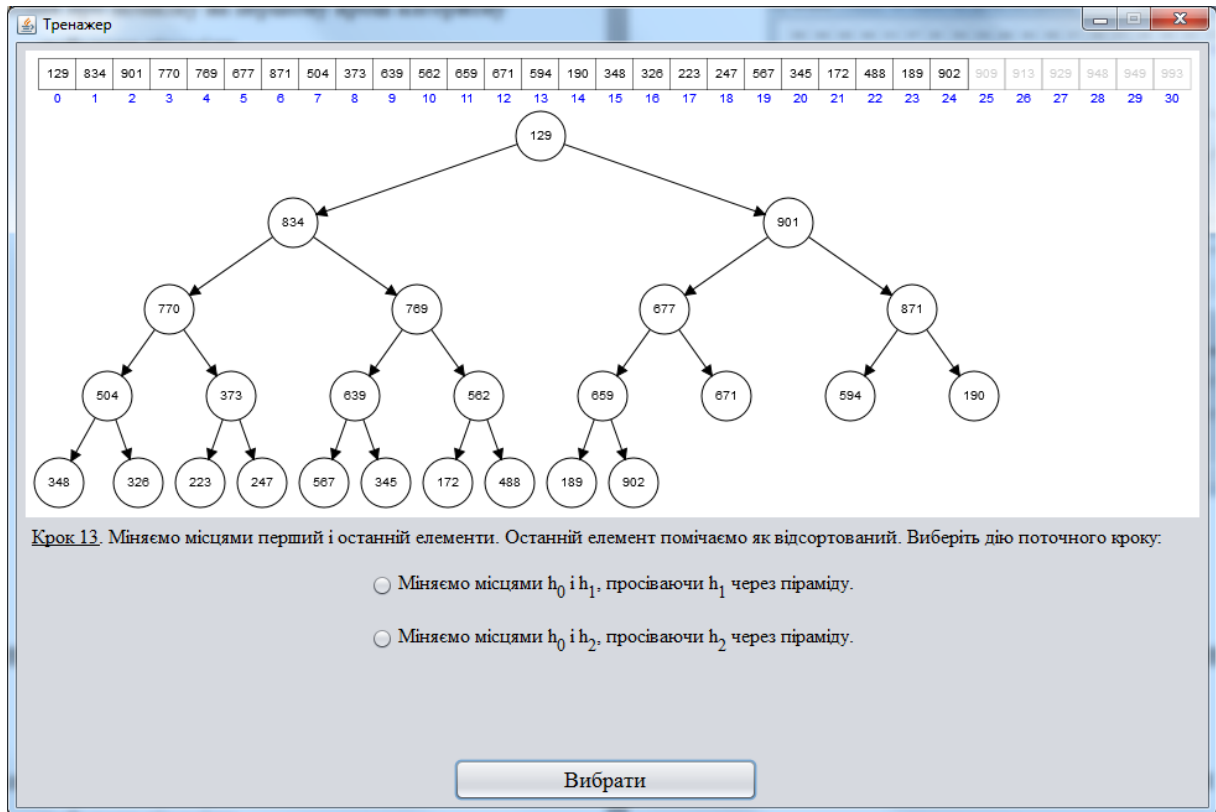


Рисунок 4.12 – Тринадцятий крок алгоритму сортування

Якщо вибрати невірну відповідь, то виведеться наступне повідомлення (рис. 4.13).

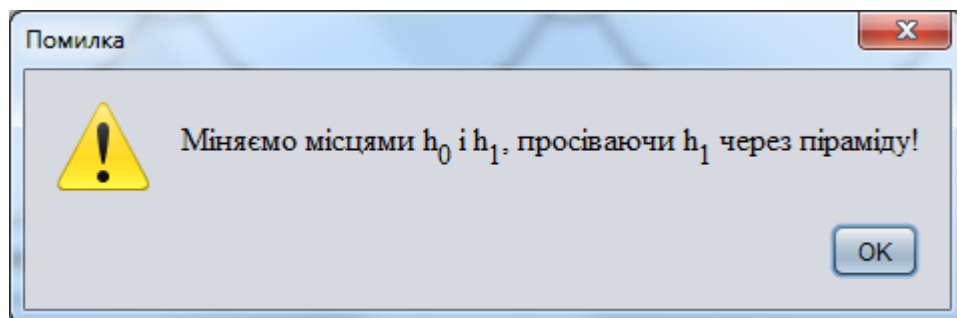


Рисунок 4.13 – Повідомлення про помилку на тринадцятому кроці алгоритму сортування

Коли запропоновану послідовність відсортовано пропонується:

- Повернутися на головну сторінку;
- Завершити роботу.

Якщо вибрано «Повернутися на головну сторінку» – відобразиться головна панель тренажеру.

Якщо «Завершити роботу» – тренажер закриється.

## ВИСНОВКИ

Програмно-педагогічні засоби дають можливість кожному студенту незалежно від рівня його підготовки брати активну участь у навчальному процесі, індивідуалізувати свій процес навчання, здійснювати самоконтроль, бути не пасивним спостерігачем, а активно отримувати знання та оцінювати свої навчальні можливості. З іншого боку, незважаючи на суттєве збільшення кількості таких програмних продуктів, їх якість та, відповідно, ефективність залишаються на недостатньо високому рівні.

Було виконано такі завдання:

- розглянуто роботи з аналогічним завданням;
- вказано позитивні аспекти оглянутих робіт;
- вказано вади розробок з оглянутих робіт;
- розглянуто матеріал за темою роботи;
- розроблено алгоритм тренажеру з теми «Пірамідальне сортування»;
- обґрунтовано вибір програмних засобів для реалізації програми;
- розроблено тренажер;
- описано створену програмну реалізацію;
- описано роботу тренажеру.

За посиланням <http://www.cs.usfca.edu/~galles/visualization/HeapSort.html> розглянуто візуалізацію пірамідального сортування. Його було використано для розробки алгоритму тренажеру.

В алгоритмі поетапно розписуються кроки пірамідального сортування. Спочатку потрібно побудувати піраміду, а вже потім провести сортування.

Якщо відповісти не вірно на будь-яке питання, то виводиться відповідне повідомлення. Воно містить таку інформацію, щоб користувач зрозумів де було допущено помилку.

На головній панелі тренажеру знаходиться:

- тема;
- назва дистанційного курсу;
- прізвище, ініціали розробника;
- прізвище, ініціали наукового керівника;
- перехід до теоретичного матеріалу;
- перехід до умови задачі й подальшого розв'язання;
- вибір мови.

Користувачу надається можливість перемкнути мову з української на англійську і навпаки, оскільки він може бути використаний при викладанні дисципліни «Алгоритми та структури даних» іноземним студентам.

При програмній реалізації спочатку було створено змінні, які будуть використовуватися.

Для кожної кнопки у тренажері розроблено свою подію, яка буде виконуватися при натисненні кнопки.

## СПИСОК ЛІТЕРАТУРИ

1. Ємець О. О. Методичні рекомендації до виконання дипломної роботи для студентів ступеня магістра спеціальності 122 «Комп'ютерні науки» / О.О.(Олег) Ємець. – Полтава : РВВ ПУЕТ, 2018. – 35 с.
2. Data Structure Visualizations [Візуалізації структур даних університету Сан-Франциско]. - [Електронний ресурс]. - Режим доступу: <https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>
3. Sorting Algorithm Animation [Сортування алгоритмів - анімація]. - [Електронний ресурс]. - Режим доступу: <http://www.sorting-algorithms.com/>.
4. Шаров Сергій Розробка програмного комплексу навчальних тренажерів з дисципліни «Архітектура ЕОМ» / Сергій Шаров, Ірина Земляна // Наукові записки: Проблеми методики фізико-математичної і технологічної освіти. – Випуск 7. – 2015. - Режим доступу: <http://eprints.mdpu.org.ua/view/year/>
5. Соколов О. Ю. Інформатика для інженерів / О. Ю. Соколов, І. Т. Зарецька, Г. М. Жолткевич, О. В. Ярова. – Харків: Факт, 2006. – 424 с.
6. Ахо А. Структуры данных и алгоритмы / А. Ахо, Дж. Хопкрофт, Дж. Ульман. – М.: Изд. дом «Вильямс», 2003. – 384 с.
7. Кнут Д. Искусство программирования, том 3. Сортировка и поиск / Д. Кнут. – М.: Изд. дом «Вильямс», 2000. – 824 с.
8. Java [Електронний ресурс] // Матеріал з Вікіпедії — вільної енциклопедії. - Режим доступу: <https://uk.wikipedia.org/wiki/Java>
9. NetBeans [Електронний ресурс] // Матеріал з Вікіпедії — вільної енциклопедії. - Режим доступу: <https://uk.wikipedia.org/wiki/NetBeans>
10. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання: ДСТУ 7.1-2006. – [Чинний від 2007-07-01]. – К. : Держспоживстандарт України, 2007. – 47 с.

**ДОДАТОК А. КОД ПРОГРАМИ**